Міністерство освіти та науки

Прикарпатський національний університет

Фізико-технічний факультет

Кафедра комп’ютерної інженерії та електроніки

Лабораторна робота № 1

З курсу “Захист інформації у комп’ютерних мережах та системах”

Виконав студент групи КІ-41

Воробій Віталій

Івано-Франківськ 2020

5 варіант

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер варіанту | Метод шифрування | Таблиця | Номер завдання в таблиці | Представлення початкового тексту |
| 5 | Підстановка | 2 | 4 | ASCII-код |

Виконання роботи

A C  
B D  
C A  
D B  
E H  
F I  
G J  
H E  
I F  
J G  
K O  
L P  
M Q  
N R  
O K  
P L  
Q M  
R N  
S U  
T V  
U W  
V :  
W S  
X T  
Y Z

Z   
 X  
. Y  
, ;  
! ?  
: -  
; .  
? ,  
- !

Рисунок 1. Вміст файлу для реалізації завдання

Написав клас для читання з файлу таблиці символів у вигляді хеш-таблиці.

package solution;  
  
import java.io.File;  
import java.io.FileNotFoundException;  
import java.io.FileReader;  
import java.util.HashMap;  
import java.util.Map;  
import java.util.Scanner;  
  
public class CharHashMapReader {  
 private final String fileName;  
  
 public CharHashMapReader(String fileName) {  
 this.fileName = fileName;  
 }  
  
 public Map<Character, Character> retrieveMap() throws FileNotFoundException {  
 Map<Character, Character> map = new HashMap<>();  
 Scanner scanner = new Scanner(new FileReader(new File(fileName)));  
 while (scanner.hasNext()) {  
 String line = scanner.nextLine();  
 if (!line.equals(""))  
 map.put(line.charAt(0), line.charAt(2));  
 }  
 return map;  
 }  
  
}

Написав клас для реалізації алгоритму шифрування підстановкою.

package solution;  
  
import java.util.HashMap;  
import java.util.Map;  
  
public class Encoder {  
 private final Map<Character, Character> charMap;  
 private Map<Character, Character> reversedCharMap;  
  
 public Encoder(Map<Character, Character> charMap) {  
 this.charMap = charMap;  
 initReversedCharMap();  
 }  
  
 private void initReversedCharMap() {  
 Map<Character, Character> reversedCharMap = new HashMap<>();  
 for (Map.Entry<Character, Character> entry : charMap.entrySet()) {  
 reversedCharMap.put(entry.getValue(), entry.getKey());  
 }  
 this.reversedCharMap = reversedCharMap;  
 }  
  
 public String cipherText(String inputText) {  
 return getProcessedString(inputText, charMap);  
 }  
  
 public String decipherText(String inputText) {  
 return getProcessedString(inputText, reversedCharMap);  
 }  
  
 private String getProcessedString(String inputText, Map<Character, Character> map) {  
 final int N = inputText.length();  
 StringBuilder builder = new StringBuilder();  
 for (int i = 0; i < N; i++) {  
 char c = inputText.charAt(i);  
 builder.append(map.getOrDefault(c, c));  
 }  
 return builder.toString();  
 }  
  
}

Клас для демонстрації роботи програми.

package solution;  
  
import java.io.FileNotFoundException;  
import java.util.Map;  
import java.util.Scanner;  
  
public class EncoderText {  
  
 public static void main(String[] args) throws FileNotFoundException {  
 *// Читання вхідної таблиці* CharHashMapReader reader = new CharHashMapReader("4\_map.txt");  
 Map<Character, Character> map = reader.retrieveMap();  
  
 Encoder encoder = new Encoder(map);  
  
 Scanner scanner = new Scanner(System.*in*);  
  
 while (scanner.hasNext()) {  
 String inputText = scanner.nextLine();  
 if (inputText.equals("stop")) {  
 break;  
 }  
 String ciphered = encoder.cipherText(inputText);  
 String deciphered = encoder.decipherText(ciphered);  
 System.*out*.println("Вхідний текст: " + inputText);  
 System.*out*.println("Зашифрований: " + ciphered);  
 System.*out*.println("Розшифрований: " + deciphered);  
 System.*out*.println("Розшифрування пройшло успішно " + ((inputText.equals(deciphered)) ? "Так" : "Ні"));  
 }  
  
 System.*out*.println("Програму завершено");  
  
  
 }  
  
}

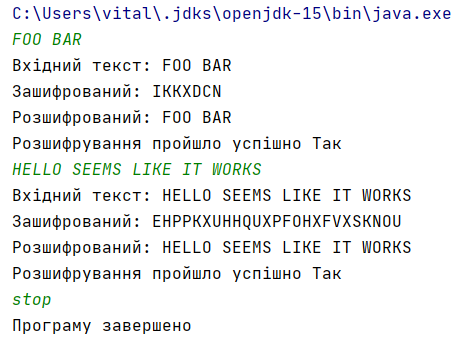


Рисунок 2. Демонстрація роботи програми

Висновок: на цій лабораторній роботі я навчився реалізовувати алгоритм підстановкою.